Research Presentation

Universal Few Shot Learner for Dense Vision Tasks

Jiho Choi

17choijiho@gm.gist.ac.kr

Electrical Engineering and Computer Science,

Gwangju Institute of Science and Technology

November 3, 2023

Machine Vision

Definition

- **machine vision** encompasses all industrial and non-industrial applications in which a combination of hardware and **software** provide operational guidance to devices in the execution of their functions based on the capture and **processing of images**.

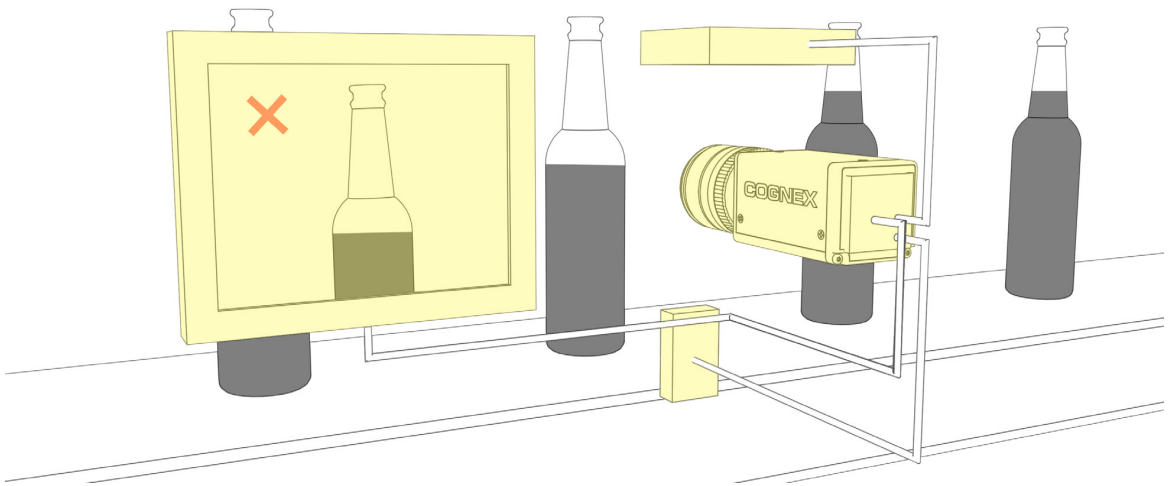


Figure: Bottle fill-level inspection example, which characterizes a binary system(e.g., Pass/Fail)[1]

[1] References) www.cognex.com

Machine Vision

Combination with Deep Learning

AI excels at addressing complex surface and cosmetic defects, like scratches and dents on parts that are turned, brushed, or shiny. [2]

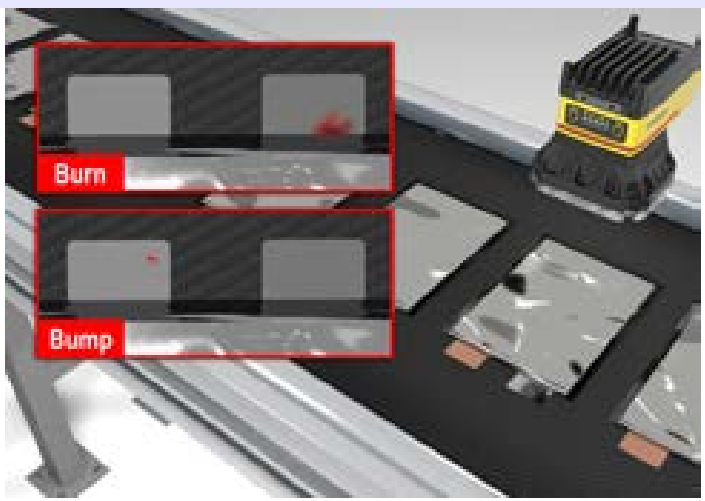


Figure: Battery Tab defect inspection[2]

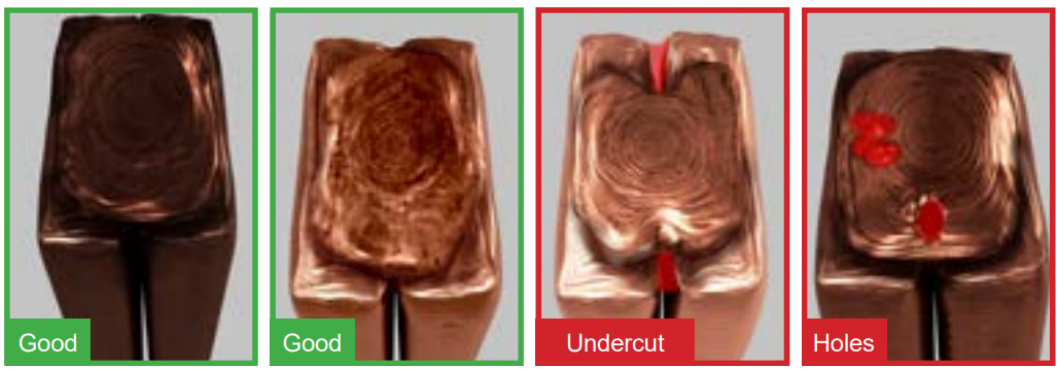


Figure: Spot Welding Inspection[2]

[2] References) www.cognex.com

Deep learning

Insufficient amount of data issue

- Generally, training deep learning models requires **lots of data**.

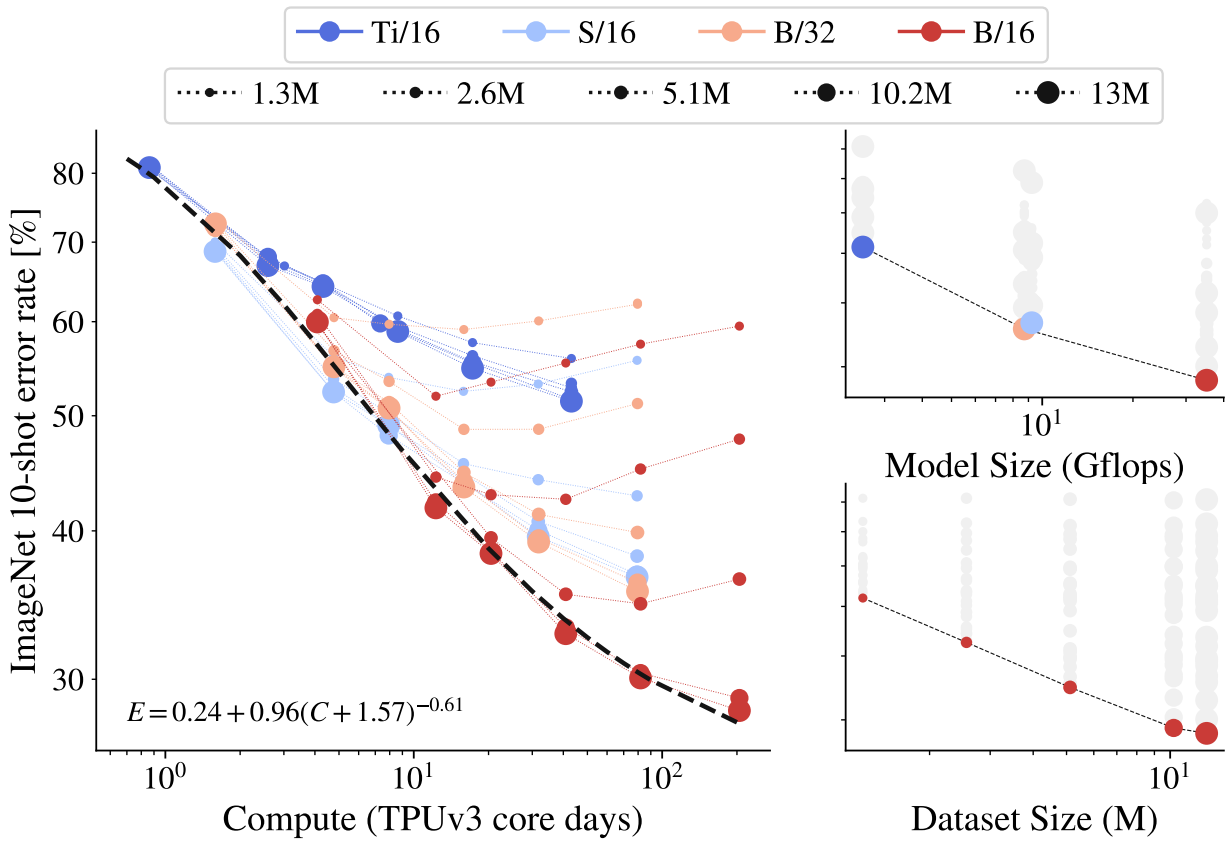


Figure: Representation quality according to model size and dataset size[10].

Task Definition

Dense prediction tasks

- Any arbitrary task $\mathcal{T}$ can be expressed as follows
- $\mathcal{T} : \mathbb{R}^{H \times W \times 3} \rightarrow \mathbb{R}^{H \times W \times \mathbf{C}_\mathcal{T}}, \mathbf{C}_\mathcal{T} \in \mathbb{N}$
- e.g., **semantic segmentation**, depth estimation, surface normal prediction, edge prediction, etc.

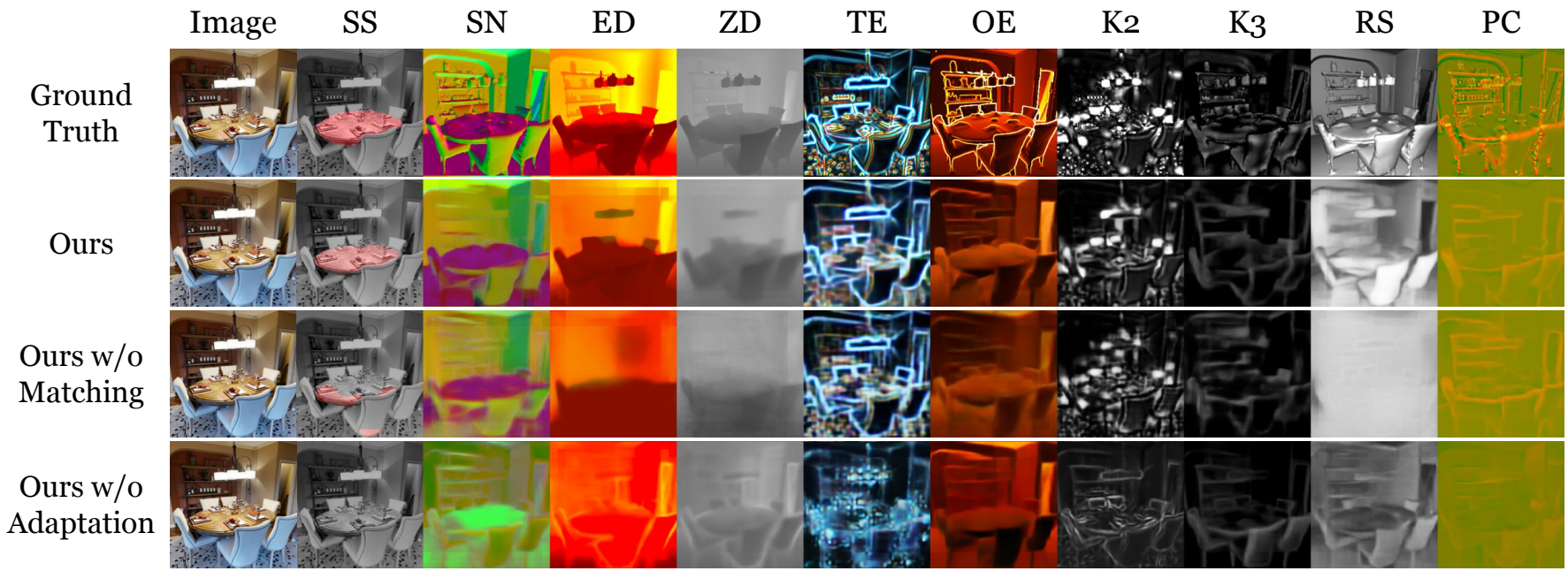


Figure: Visualization of Taskonomy[9]'s dense vision tasks

Task Definition

Universal Few-shot Learner $\mathcal{F}$[5]

- A universal few-shot learner $\mathcal{F}$, for any such task $\mathcal{T}$, can produce predictions $\hat{Y}^q$ for an unseen image(query) X^q given a few labeled examples(support set) $\mathcal{S}_\mathcal{T}$
- That is, $\widehat{\mathbf{Y}}^{\mathbf{q}} = \mathcal{F}(\mathbf{X}^{\mathbf{q}}; \mathcal{S}_\mathcal{T}), \mathcal{S}_\mathcal{T} = \{(\mathbf{X}^{\mathbf{i}}, \mathbf{Y}^{\mathbf{i}})\}_{\mathbf{i} \leq \mathbf{N}}$
- Conventional episodic training was adopted for the universal few-shot learner $\mathcal{F}$
  - e.g., at each **meta-training** iteration 4 shot, 5 tasks
  - e.g., at each **meta-test** iteration 10 shot, 1 task

Task Definition

"Universal" Few-shot Learner $\mathcal{F}$

- **The task-agnostic model** $\mathcal{F}$ for all dense vision tasks $\mathcal{T}$.
- The backbone networks for the image encoder and the label encoder are **vision transformers**[2].

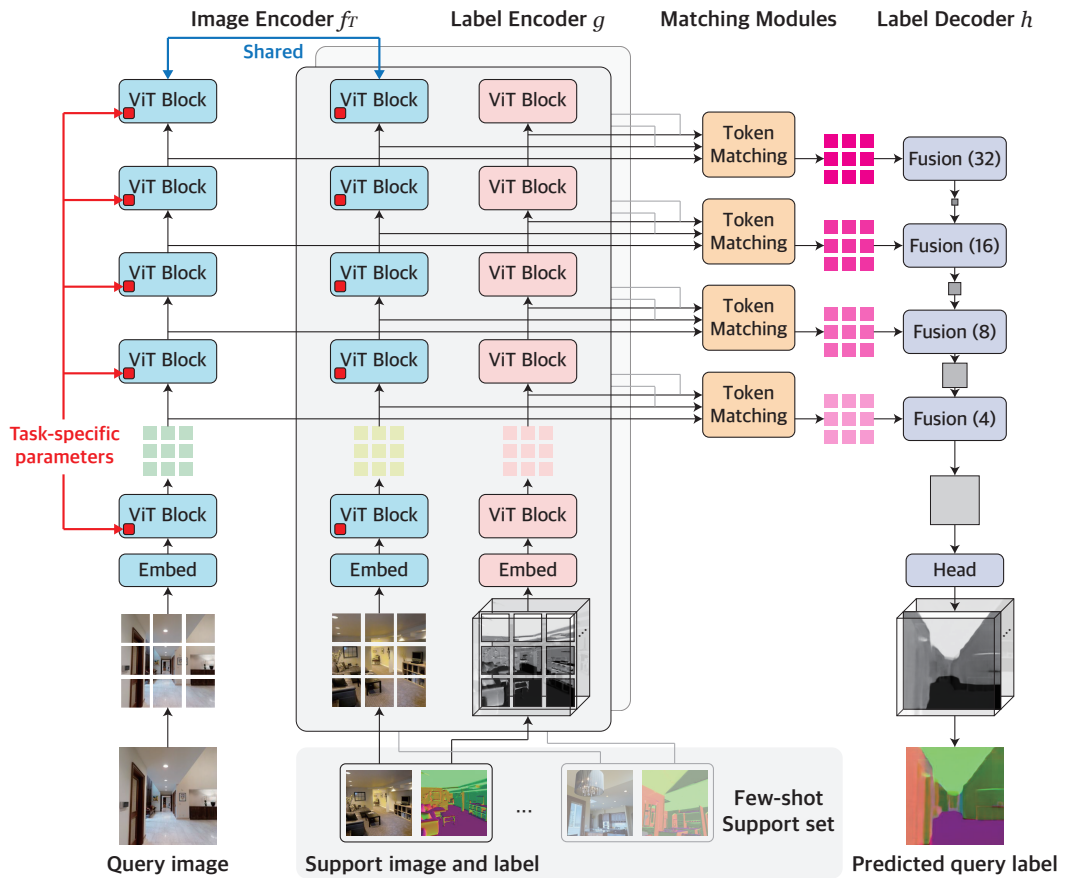


Figure: Model Structure for the universal few-shot learner $\mathcal{F}$ from [5]

Task Definition

Universal "Few-shot" Learner $\mathcal{F}$

- **with** $< 0.004\%$ of the full supervision(10 labeled images), the $\mathcal{F}$ shows the similar performances of **fully-supervised** models(e.g., InvPT) on some tasks.
- by inscreasing size of support set **from** $< 0.004\%$ **to** $< 0.1\%$, the $\mathcal{F}$ shows similar or **better performances** than fully-supervised methods.

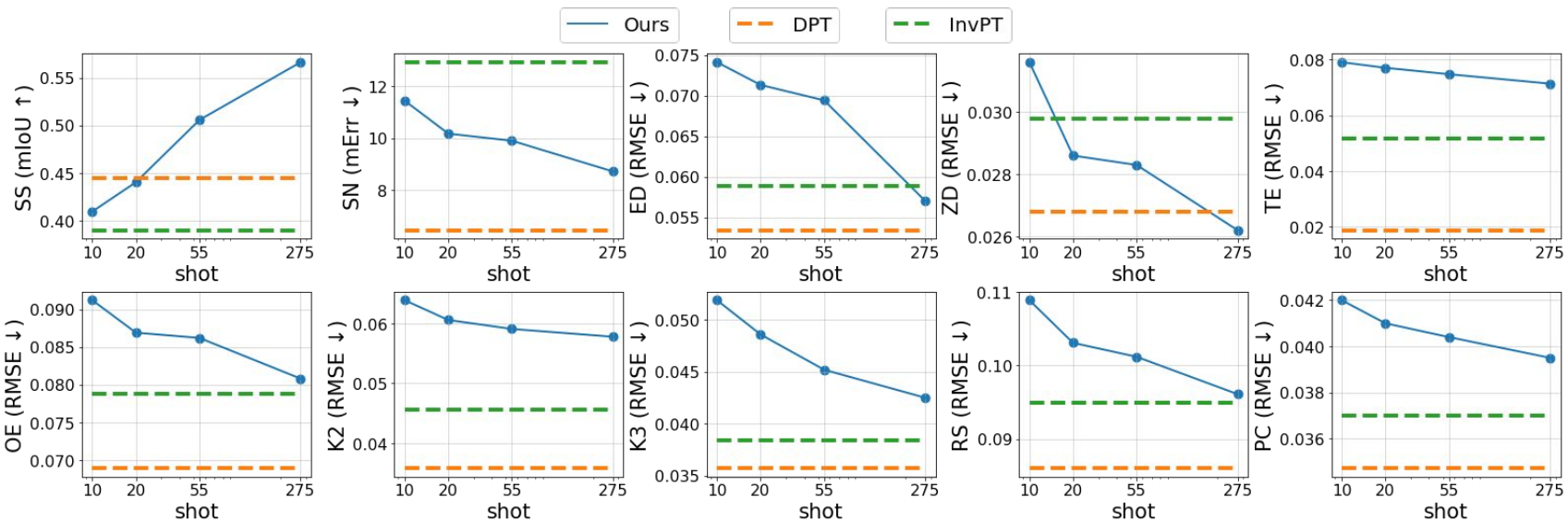


Figure: Performance of the $\mathcal{F}$ on various shots from [5]

Prompting

Viusal Prompt Tuning[3]

- **Prompting**: prepending language instruction to the input text so that a pre-trained LM[a] can "understand" task.
- **Prompt Tuning**: treats the prompts as **task-specific continuous vectors** ans directly **optimize** them via gradients **during fine-tuning**.
- **Visual Prompt Tuning**: introduces only **a small amount of trainable parameters**[b] in the **input space** while keeping the model backbone **frozen**

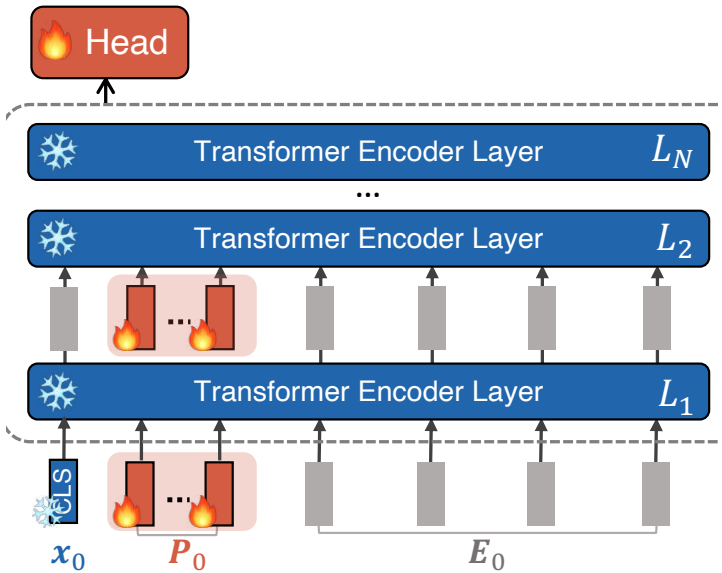


Figure: Visual Prompt Tuning Method from [3]

[a] i.e., language model, e.g, BERT, GPT, etc.
[b] i.e., 0.53% of backbone network

Visual Prompt Tuning

Quantiative Results

- Experiments on various prompt schemes[3, 1, 8] are conducted.
- The **well-structured** prompt scheme[1] works well on some tasks.

Supervision	Model	Tasks									
		Fold 1		Fold 2		Fold 3		Fold 4		Fold 5	
		SS	SN	ED	ZD	TE	OE	K2	K3	RS	PC
		mIoU ↑	mErr ↓	RMSE ↓	RMSE ↓	RMSE ↓	RMSE ↓	RMSE ↓	RMSE ↓	RMSE ↓	RMSE ↓
Full	DPT	**0.4449**	**6.4414**	**0.0534**	**0.0268**	**0.0188**	**0.0689**	**0.0358**	**0.0357**	**0.0860**	**0.0347**
10-Shot ($< 0.004\%$)	Paper	0.4097	11.4391	**0.0741**	**0.0316**	**0.0791**	**0.0912**	0.0639	0.0519	**0.1089**	**0.0420**
	Reproduced	**0.4328**	10.2526	0.0875	0.0334	0.0833	0.0995	0.0630	0.0496	0.1167	0.0433
	vpt-deep[3]	0.2230	12.1731	0.1348	0.0471	0.0881	0.1183	0.0628	0.0520	0.2079	0.0453
	vpt-express-bias[4]	0.3820	**10.1446**	0.0902	0.0358	0.0792	0.1019	**0.0594**	**0.0491**	0.1219	0.0435

[3] the number of prompt is 20
[4] the number of prompt is 5

Image, Label Encoders

Image encoder: BEIT[7]

- Directly using pixel-level auto-encoding for vision pre-training pushes the model to focus on **short-range dependencies and high-frequency details**.
- BEIT overcomes the above issue by predicting discrete visual tokens, which summarizes the details to **high-level abstractions**.

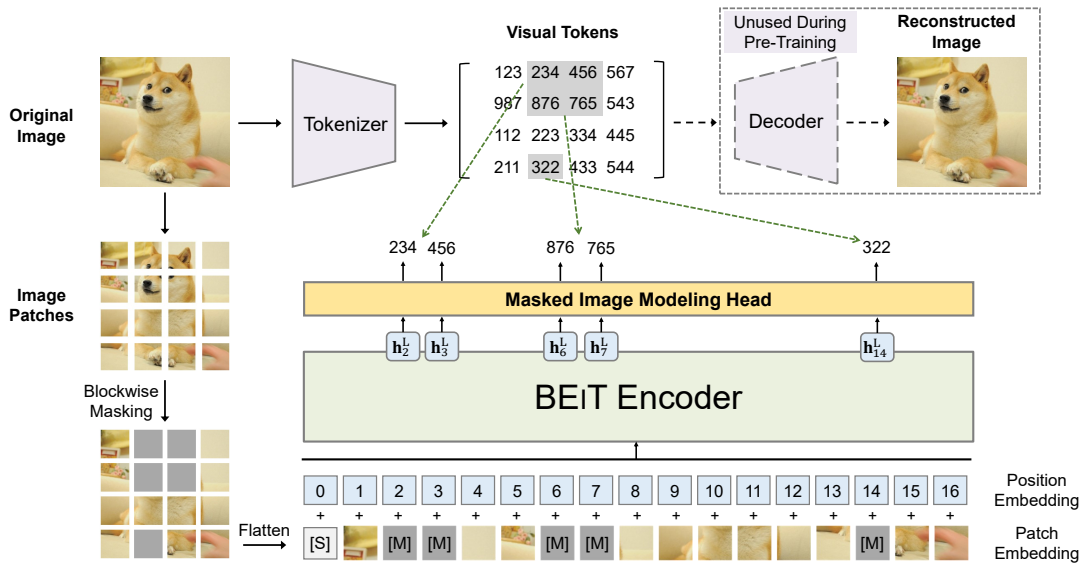


Figure: BEIT[7]'s overall pipeline

Image, Label Encoders

Label encoder: SAM[6]

- Uses Segment Anything's Image Encoder as Label Encoder in [5]
- which is MAE pre-trained and then trained on SA-1B(11M images, 1B annotations)
- To albeit its capability on semantic segmentation and edge detection tasks

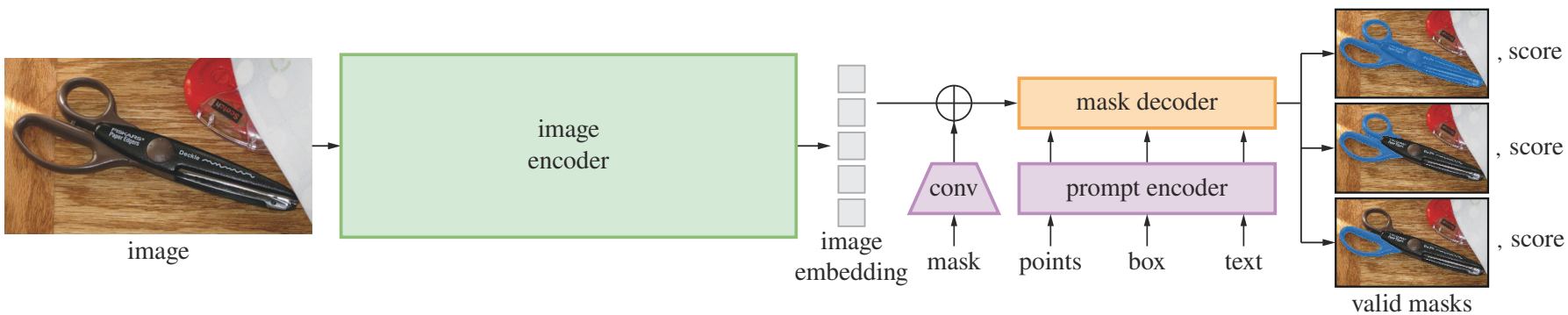


Figure: SAM[6]'s overall pipeline

Sum up

Prompt Transfer, Encoders

- image, label encoder 모두 meta-train 단계에서 optmization을 거치나,
- 이 둘의 pre-trained weight가 학습될 때(i.e., BEIT, SAM) 방법의 차이가 크기에 이들의 representation space상에서의 feature representation의 차이가 있을 것으로 가정함
- 따라서, meta-test 단에서 10-shot optmization을 거칠 때, 두 encoder 간에 information communication을 위하여 prompt transfer module을 고안함

Prompt Transfer

Methods: MaPLe[4]

- We reason that in prompt tuning it is essential to take **a multi-modal approach and simultaneously adapt both the vision and language branch of CLIP** to achieve completeness in context optimization.

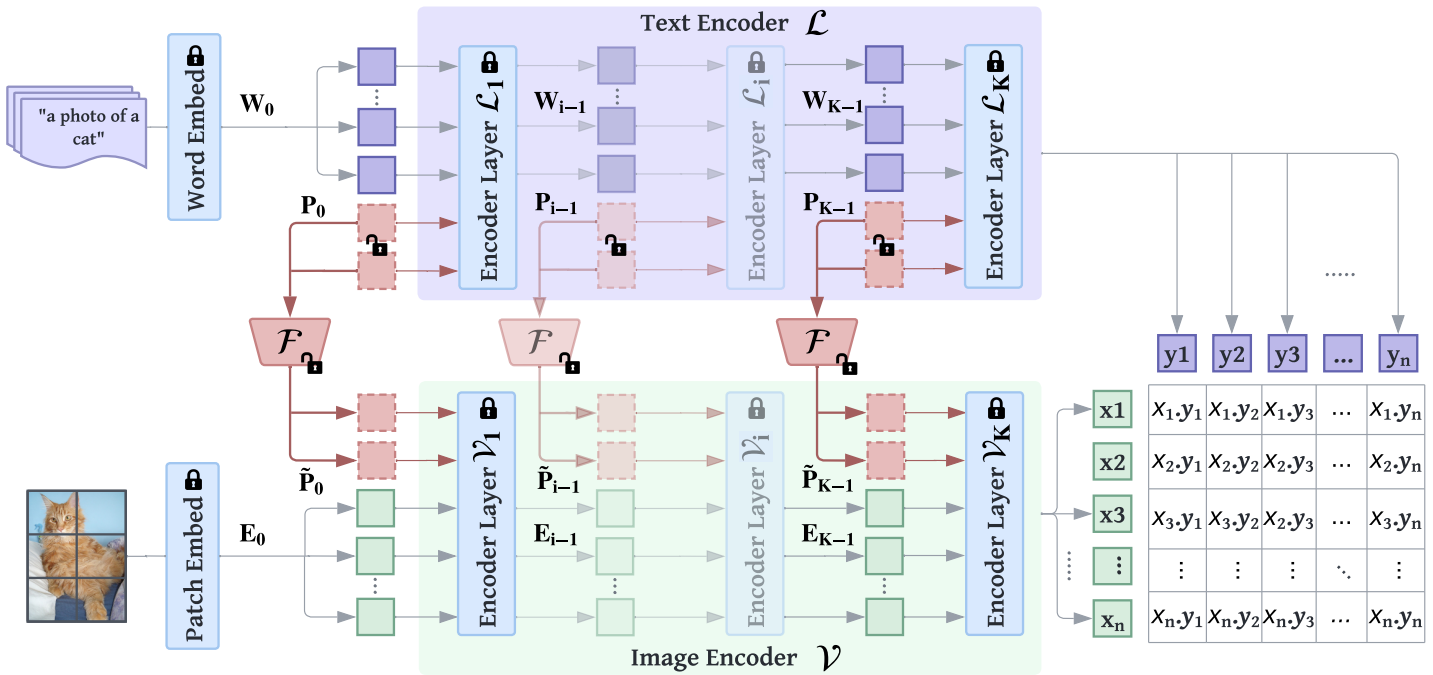
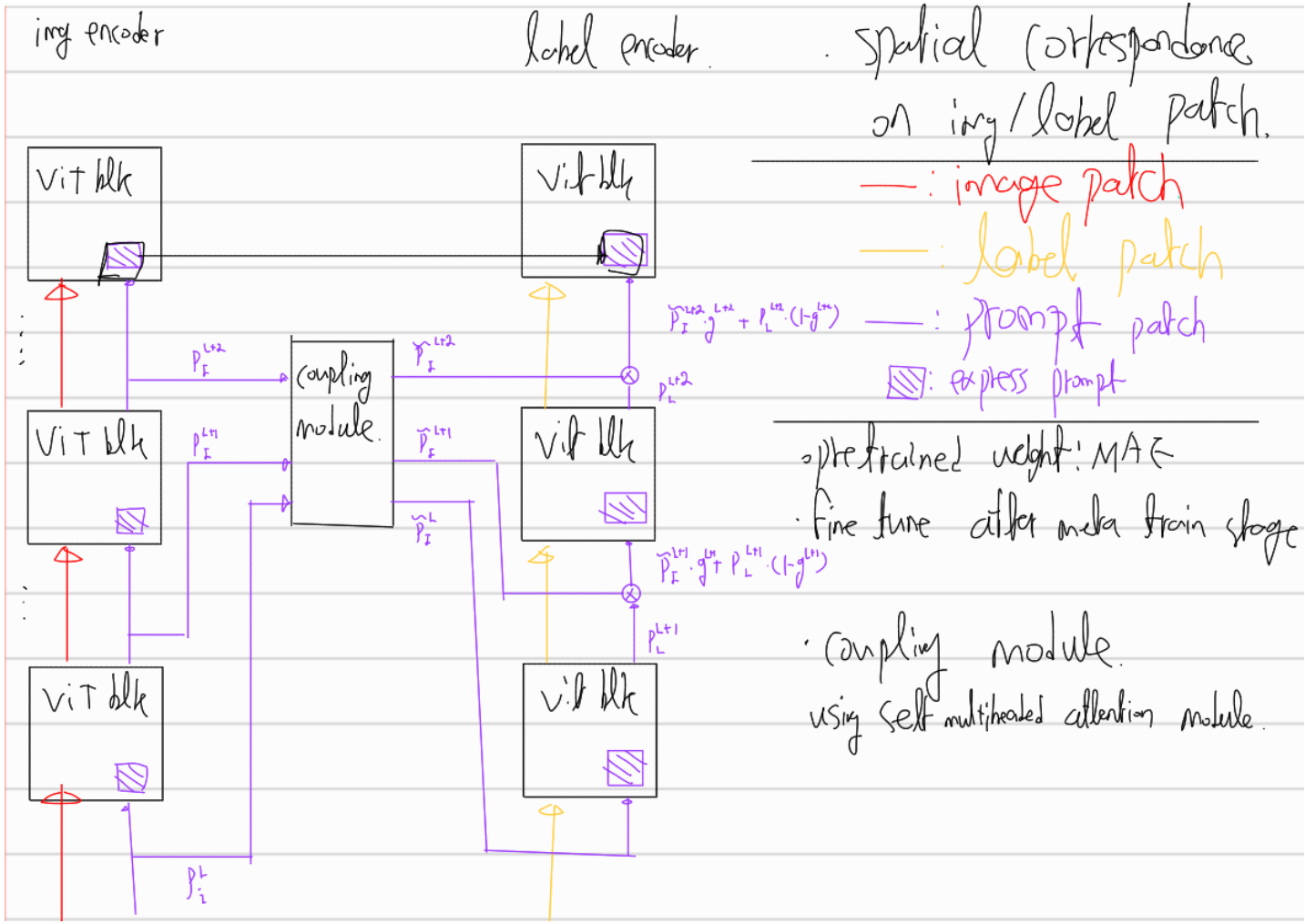


Figure: Multi-modal Prompt Learning[4]

Prompt Transfer

Joint Encoders with Coupled ExPRes[1] Prompts



Experiments Results

Prompt Transfer, Encoders

Task	Prompt Transfer Methodology	Result	비고
Semantic segmentation	Transformer layer	0.459852(mIOU ↑)	0.4449(full supervision)
Surface Normal	Transformer layer	10.74502(mErr ↓)	11.43(VTM paper), 10.01(ExPres)

Table: Experiments results on two tasks

References I

[1] Rajshekhar Das et al. *Learning Expressive Prompting With Residuals for Vision Transformers*. Mar. 2023. DOI: 10.48550/arXiv.2303.15591. arXiv: 2303.15591 [cs]. (Visited on 06/29/2023).

[2] Alexey Dosovitskiy et al. *An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale*. June 2021. arXiv: 2010.11929 [cs]. (Visited on 07/25/2023).

[3] Menglin Jia et al. *Visual Prompt Tuning*. July 2022. DOI: 10.48550/arXiv.2203.12119. arXiv: 2203.12119 [cs]. (Visited on 06/29/2023).

[4] Muhammad Uzair Khattak et al. *MaPLe: Multi-modal Prompt Learning*. Apr. 2023. DOI: 10.48550/arXiv.2210.03117. arXiv: 2210.03117 [cs]. (Visited on 06/27/2023).

References II

[5] Donggyun Kim et al. *Universal Few-shot Learning of Dense Prediction Tasks with Visual Token Matching*. Mar. 2023. DOI: 10.48550/arXiv.2303.14969. arXiv: 2303.14969 [cs]. (Visited on 06/29/2023).

[6] Alexander Kirillov et al. *Segment Anything*. Apr. 2023. DOI: 10.48550/arXiv.2304.02643. arXiv: 2304.02643 [cs]. (Visited on 07/16/2023).

[7] Zhiliang Peng et al. *BEiT v2: Masked Image Modeling with Vector-Quantized Visual Tokenizers*. Oct. 2022. DOI: 10.48550/arXiv.2208.06366. arXiv: 2208.06366 [cs]. (Visited on 09/18/2023).

References III

[8] Seungryong Yoo et al. *Improving Visual Prompt Tuning for Self-supervised Vision Transformers*. June 2023. arXiv: 2306.05067 [cs]. (Visited on 07/26/2023).

[9] Amir Zamir et al. *Taskonomy: Disentangling Task Transfer Learning*. Apr. 2018. DOI: 10.48550/arXiv.1804.08328. arXiv: 1804.08328 [cs]. (Visited on 07/04/2023).

[10] Xiaohua Zhai et al. *Scaling Vision Transformers*. June 2022. arXiv: 2106.04560 [cs]. (Visited on 08/11/2023).

광주과학기술원

Gwangju Institute of Science and Technology

VTM model

Task-Agnostic Architecture

- A unified architecture that can handle all dense prediction tasks.
- It does not exploit any kind of **prior knowledge or inductive bias** specific to certain tasks according to the author.

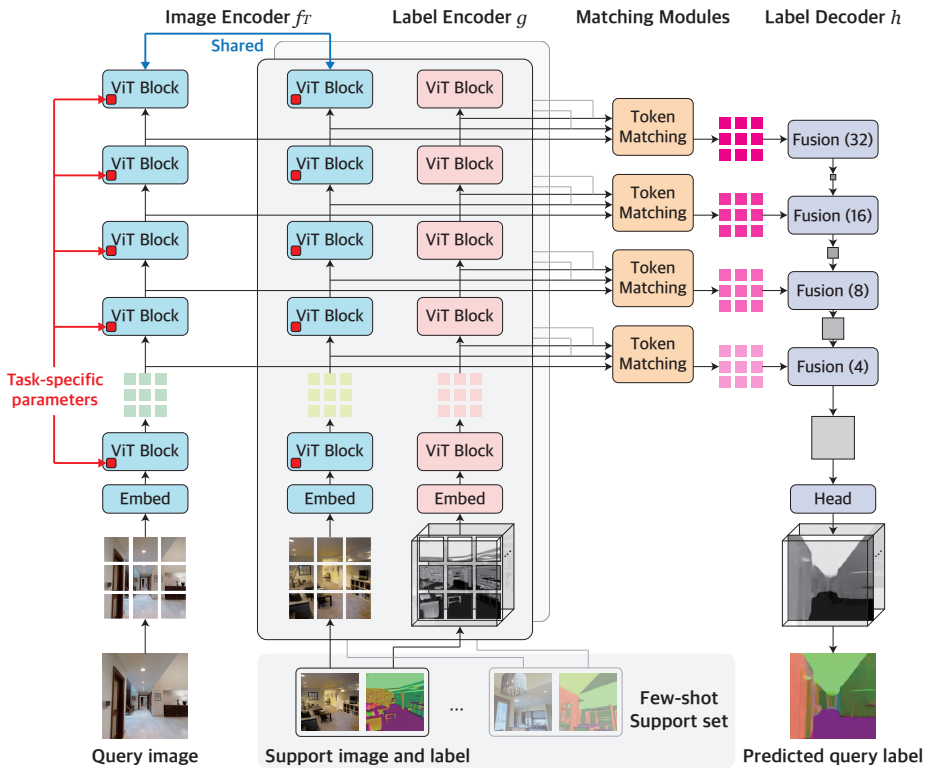


Figure: Model Structure for the universal few-shot learner $\mathcal{F}$

VTM model

Task-Agnostic Architecture

- **Image Encoder**
  - Vision Transformer initialized by pretrained BEiT(e.g., self supervised).
  - Tokens extracted from intermediate ViT blocks to form hierarchical feature.
  - **shared parameters θ, task-specific parameters $\theta_{\mathcal{T}}$**
  - **Bias Tuning** on $\theta_{\mathcal{T}}$ at meta test period.
- **Label Encoder**
  - Vision Transformer initialized randomly and trained from scratch
  - Tokens extracted from intermediate ViT blocks to form hierarchical feature.
  - **shared parameters only**
- **Label Decoder**
  - adopted decoder architecture of Dense Prediction Transformer
  - trained from scratch

VTM model

Visual Token Matching(VTM) Module

- **non parametric approach** that operates on patches, where the query label is obtained by **weighted combination of support labels**.
- $X = \{\mathbf{x}_j\}_{j \le M}$ denote an image (or label) on patch grid of size $M = h \times w$, where $\mathbf{x}_j$ is j-th patch. a query image $X^q = \{\mathbf{x}_j^q\}_{j \le M}$ and a support set $\{(X^i, Y^i)\}_{i \le N} = \{(\mathbf{x}_k^i, \mathbf{y}_k^i)\}_{k \le M, i \le N}$ for a task $\mathcal{T}$.
- predict the query label $Y^q = \{\mathbf{y}_j^q\}_{j \le M}$ patch-wise by,
  $\mathbf{g}(\mathbf{y}_\mathbf{j}^\mathbf{q}) = \sum_{\mathbf{i} \le \mathbf{N}} \sum_{\mathbf{k} \le \mathbf{M}} \sigma \left(\mathbf{f}_\mathcal{T}(\mathbf{x}_\mathbf{j}^\mathbf{q}), \mathbf{f}_\mathcal{T}(\mathbf{x}_\mathbf{k}^\mathbf{i}) \right) \mathbf{g}(\mathbf{y}_\mathbf{k}^\mathbf{i})$
- where $f_\mathcal{T}(\mathbf{x}) = f(\mathbf{x}; \theta, \theta_\mathcal{T}) \in \mathbb{R}^d$ and $g(\mathbf{y}) = g(\mathbf{y}; \phi) \in \mathbb{R}^d$ correspond to the image and label encoder, respectively. $\sigma : \mathbb{R}^d \times \mathbb{R}^d \to [0, 1]$ denotes a similarity function.
- predicted label patch a label patch $\hat{\mathbf{y}}_j^q = h(g(\mathbf{y}_j^q))$ by introducing a label decoder $h \approx g^{-1}$.

VTM model

Visual Token Matching(VTM) Moudule

- **implemented by multihead attention layer**
- $MHA(q,k,v) = Concat(o_1, \ldots, o_H)w^O$
- $o_h = Softmax(\frac{qw_h^Q(kw_h^K)^T}{\sqrt{d_H}})vw_h^V$
- $\mathbf{q} \in \mathbb{R}^{M\times d}$, $\mathbf{k}, \mathbf{v} \in \mathbb{R}^{NM\times d}$ H is number of heads, d_H is head size, and $w_h^Q, w_h^K, w_h^V \in \mathbb{R}^{d\times d_H}$, $w^O \in \mathbb{R}^{Hd_H\times d}$